

The fuzzy boundary

Four products that are also services

Mike Kuniavsky

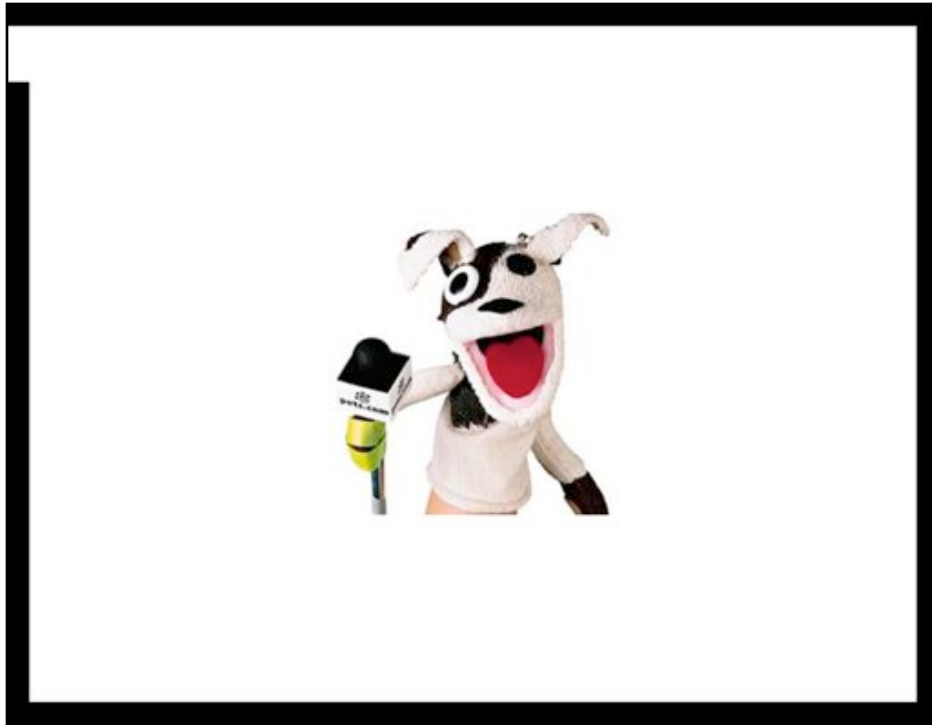
Theory and Practice of TUI

November 3, 2009

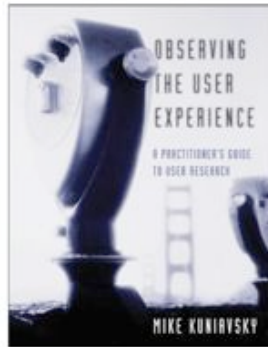
Thank you very much for inviting me, Kimiko. It's always a pleasure to speak at Berkeley and to speak to this class. Today I want to talk about how ubiquitous computing technology creates a fuzzy boundary between products and services and to look a handful of case studies that show a range of approaches for merging products and services.



But first, let me tell you a little about who I am. I'm a user experience researcher and designer. I spend much of my time thinking about how technologies and people affect each other from social, economic, historical and technological perspectives, and how the technological side of that relationship can be made better, or at least more interesting, for the human side of it.



I spent a little more than 10 years doing design and research for the web. I worked with many dotcoms, some famous, some infamous.



I sat out the first dotcom crash writing a book based on the work I had been doing. It's a cookbook of user research methods that some of you may have used in a class here at the iSchool.



In 2001 I co-founded a design and consulting company called Adaptive Path. Things went very well, Adaptive Path is doing very well, but I was interested in other ways that technology was changing society.



So I founded a company with Tod E. Kurt called ThingM to pursue these ideas commercially three years ago. We're a ubiquitous computing consumer electronics company, which sounds fancy, but we're pretty small. We design, manufacture and sell ubicomp hardware. You may have run across BlinkM, our smart LED in this class.



Today I want to talk about products...

Hammer photo CC Attribution by Cellular Immunity, <http://www.flickr.com/photos/58534808@N00/323527111>



and services.



And how pervasive networking and object tracking is fundamentally changing how we understand where a product ends and a service begins.

Photo CC Attribution-Share Alike 2.0 by Marshall Astor: <http://www.flickr.com/photos/lifeontheedge/286867230/>. Found on Flickr.



It used to be pretty easy to understand what a product was. It's a thing. It's a tangible thing that's used to achieve an effect. You buy it, then you either consume it or you keep it and use it again. If it's not a consumable, it's a tool. To get value out of a tool, for example, it must be used, but then it retains the capability of providing that value indefinitely. You can give it to someone else, or you can leave it sitting in a box for twenty years and then take it out again and it's still just as good, assuming it hasn't decayed too much.

Photo CC Attribution by Elsie esq. : <http://www.flickr.com/photos/elsie/384043919/>



Services were also relatively easy to define. A service is an effect that someone else does for you. You pay for it once and the service is performed once, or you get a subscription and the service is performed regularly. What you own is not a product for achieving the effect yourself, but an agreement that someone will perform the service on your behalf. The service may not provide the flexibility of a specific product, but the value is that it does not require acquiring special knowledge or taking time to achieve the effect. For some things, say accounting, you don't actually have to be present for the service to be performed on your behalf. You pay for it, you trigger it to start, and do other things while it's happening in the background.

Photo CC Attribution-Share Alike by blmurch.

<http://www.flickr.com/photos/blmurch/338094141/>



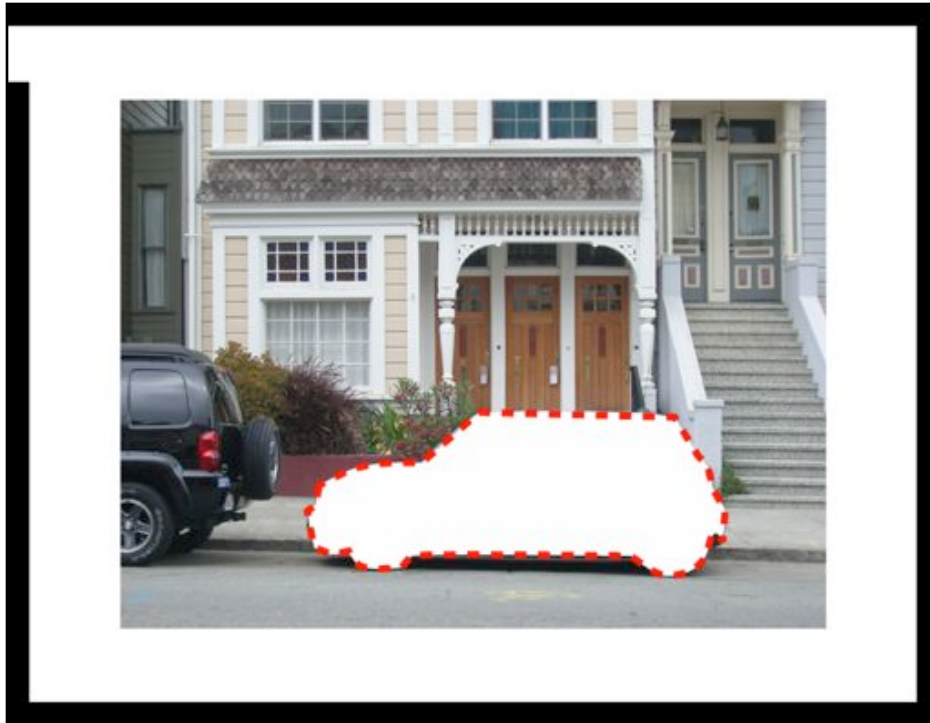
But now this distinction is being eroded. Now it's possible to have objects that provide some of their value based on what can be done with them locally, acting like product, and provide some of their value through work they trigger on your behalf, and act like services.

For example, you can use your iPhone as a calculator. That's a local tool that requires nothing of the iPhone's networking capabilities. You can of course use your iPhone as a map viewer, and at that point you are transferring to using the device as a service that is completely dependent on its networking. The second you bring up Google Maps, a host of services—things that your local device is not doing, but that are being done on your behalf—are triggered. Your device has moved from being a standalone product to being a representative of a service.

This quality of being able to exist simultaneously as a local product and as gateway to a networked service is almost exclusively a result of the fact that it's a digital, networked device. Before it was possible to individually identify and track unique objects, and to change their behavior remotely, products and services were not tightly coupled. Very few things could simultaneously be a product and a service.

Photo CC Attribution by jeffwilcox.

<http://www.flickr.com/photos/jeffwilcox/2657387713/>



Things that are products and services existed before digital technology, but they were either very large, very expensive or both. For example, an apartment is both a thing and a service. Digital technology has fundamentally changed this relationship, and created a completely new set of design challenges in the process.

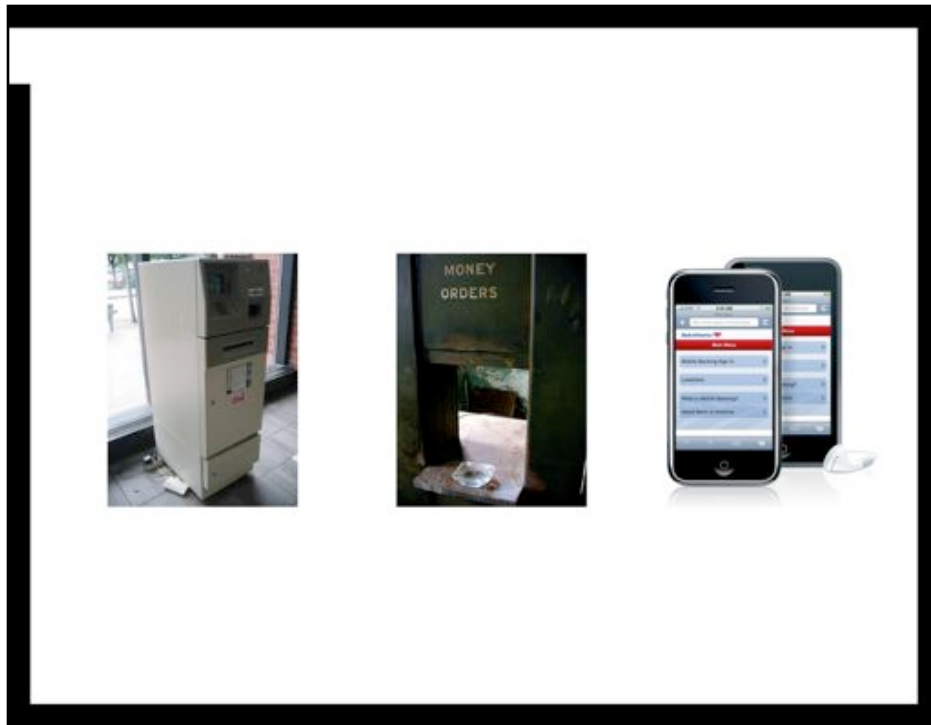
Let me give you another example.

When you buy into a car sharing service such as City Carshare or ZipCar you subscribe to a service. That service is enabled by a central network. You can only open the car and start the engine when your specific keyfob RFID is scheduled to open and start it. It uses a GPS to track where the car is, whether it's been dropped off at the right location, and how far it's been driven and an embedded computer to make decisions based on all of that information. All of that is transparent to you, the subscriber. As a result, the relationship you have with these cars is very different than either owning a car, renting a car, or taking a taxi. It succeeds because the experience of the service is much like car ownership. Subscribers have access to it 24 hours a day, 7 days a week, with very little advance notice.

However, unlike an owned car, what they own possess is an agreement for the delivery of a service of flexible, on-demand, point to point transportation. As a subscriber, you also don't own a single car, but the right to access the idea of a car, instantiated as one of a collection of cars, much as when you use Google Maps on an iPhone, you're not offered a single map, but the right to access the specific map tiles you need when you need them.



Thus, if we define a spectrum between a product and a service, City Carshare falls somewhere in the middle, probably closer to being a service than a product.



What's interesting to me is that digital technology makes it very easy for a single service to have multiple different kinds of ways it's expressed. For example, the service a bank provides is accessible through a number of different products. You may interact with a banking service through an ATM, by talking to a person, or through your phone, but it's still the same service. It's the same money.

The more tightly coupled the device is to that service, the less of a general purpose computing device it is. In many cases, the device used to access the service is useless without the service itself. An ATM that's not connected to its banking network is useless without that network. A banking application that can't connect to the banking service it represents is largely useless without it.

I describe products that are dependent on their underlying service as avatars of that service.

ATM photo CC Attribution-Share Alike redjar

<http://www.flickr.com/photos/redjar/114099266/>

Teller window CC Attribution by Todd Ehlers

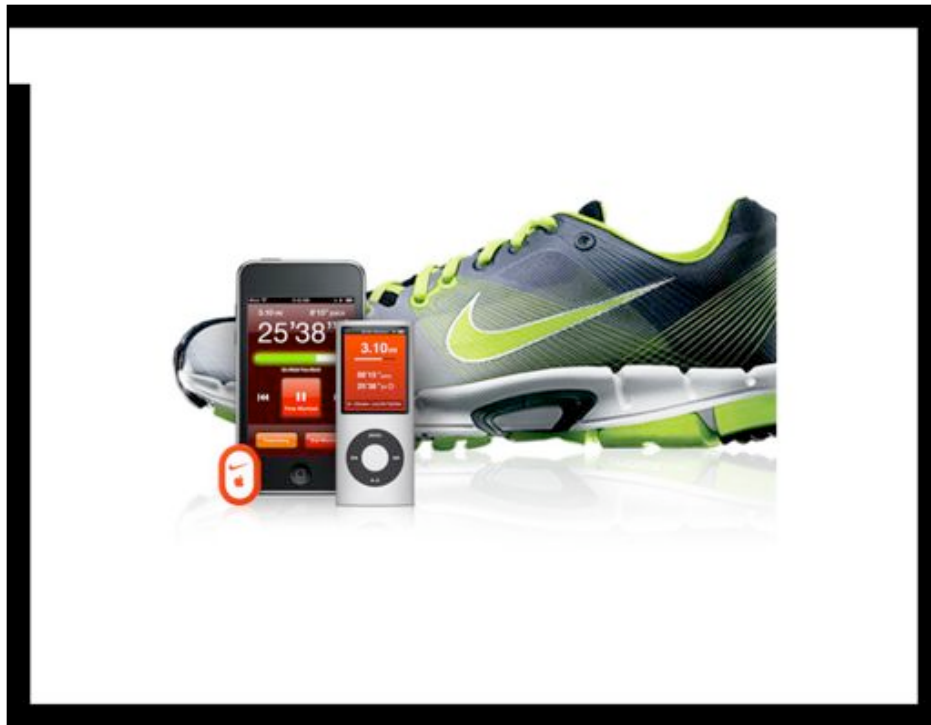
<http://www.flickr.com/photos/eklektikos/450181744/>



Now, let me explain how I'm using the term avatar. Ever since Neal Stephenson's science fiction novel *Snow Crash*, this is how people have imagined avatars: as representations of physical things, such as people, in digital spaces. I'm using it in a way that's closer to its original Hindu meaning: as physical manifestations of abstract entities. This is the goddess Devi, and she manifests in the world through nine different avatars.

Image CC Attribution-ShareAlike by World Economic Forum

<http://www.flickr.com/photos/worldeconomicforum/368896596/>



Moreover, each avatar can emphasize a certain facet of the service over the others.

The data collected by the Nike+iPod system is expressed through an iPod, an iPhone application, a desktop widget, and a Web site. Each of these avatars is tuned to emphasize a different aspect of the data. One emphasizes data collection, one emphasizes data tracking and one emphasizes data analysis.



Now I'd like to walk you through several examples of products that cross the boundary with services in an interesting way.

This is one of my favorite new tangible interaction products that blurs the line between a physical device and a service. It's called Clickables and it's a product from a Hong Kong company called TechnoSource. It's part of Disney's new Fairies initiative. It's the one that expands on the character of Tinkerbell from Peter Pan.

Each of the toys in the Clickables line connects over the internet to a Clickables service.

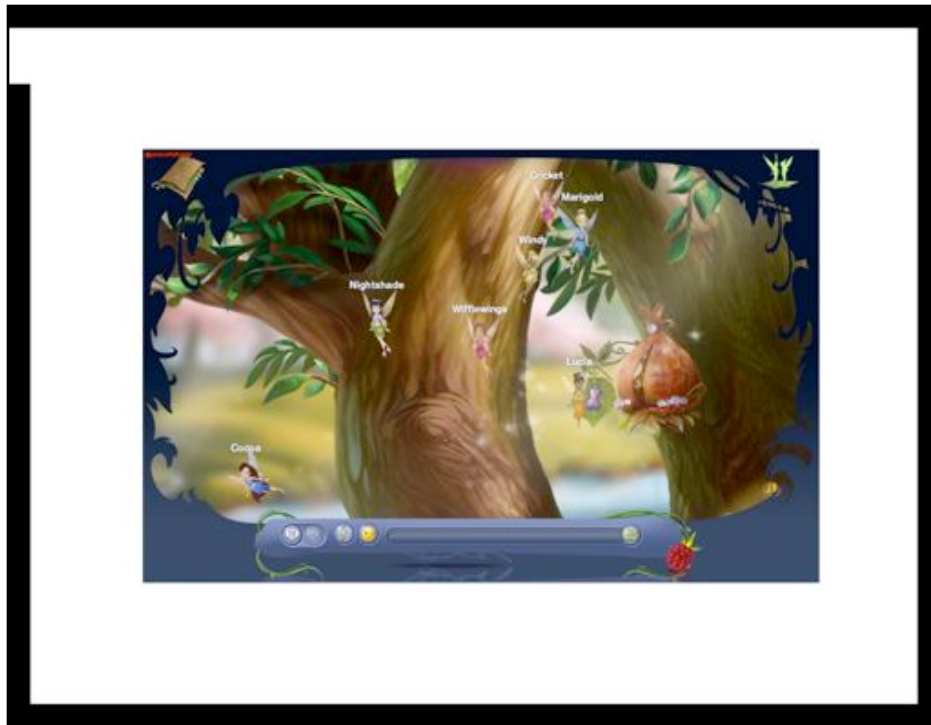
Source: Disney Clickables



Here's one of the ways it works: when two kids put their Clickables bracelets together, their avatars link up in Pixie Hollow, the online social network associated with the Fairies brand. This bridges the physical world of kids with their social network in a transparent and familiar way. All of the products in this line have such an online-offline existence.



Moreover, the linkage continues: when you get one of the charm bracelets and you touch the charms to the USB-connected jewelry box, your fairy gets a version of the same charm.



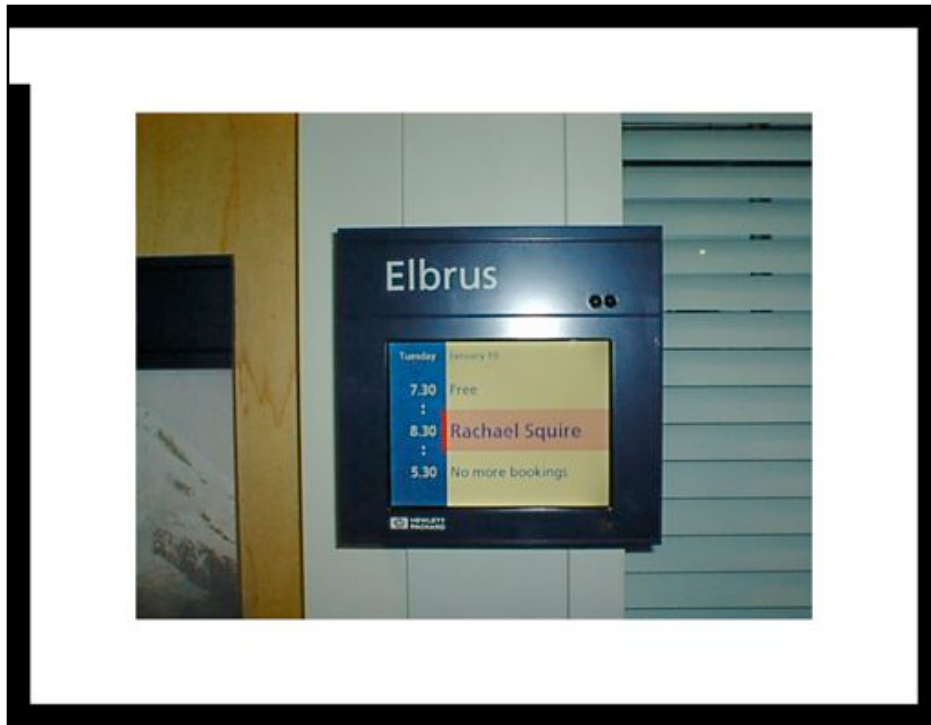
But the designers are savvy: because the digital representation of the jewelry can be replicated much more cheaply than the physical jewelry, when a character gets a jewel, it comes with a certain number of digital versions of itself so that you can then give that same jewel to a number of your online friends. Thus, the experience designers at TechnoSource really understand the social dynamic of their audience AND the potential of the technology and are exploiting both.

When you couple this with the fact that this all takes place in Disney's Fairy's universe, with all of the other media—the books, the videos, the games, the costumes, the theme parks, the toys—that are thematically tied together, you see the power of the jewelry as avatars of this very sophisticated entertainment service. The jewels by themselves evoke the imaginary world, but then they go beyond it by actually creating a kind of literal direct connection with the world. When two children touch them together, they are doing much more than just fantasizing, they are triggering a range of digital services,

If their online virtual world wasn't so unusable, I think that this would be an incredible experience. As it is, I think it really points the way toward future transmedia storytelling with incredible potential for engagement.



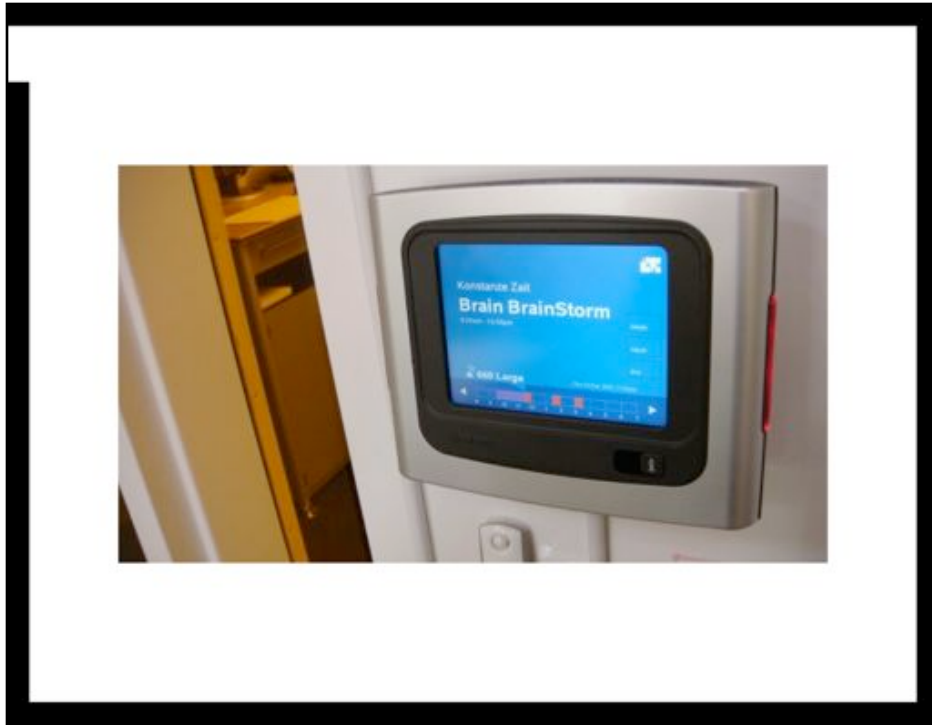
Next, I want to talk about a different product. This is the Steelcase RoomWizard, which was developed in the late 90s and has been on sale for a number of years. It's a room scheduling appliance that was designed to be simultaneously be an avatar and server for a service.



Let me give you some background on it because I believe it's an important early commercial ubiquitous computing product. It was designed more than 10 years ago in HP Labs Bristol as an experiment in what would happen when LCD panels became dirt cheap, but I would like to focus on its relatively unique position as a product that's both a service and a product.

This is an early prototype.

Photo: Simon Lewis



Because it was to be sold like furniture through Steelcase's traditional sales channels, it had to behave like furniture, even though it was at its heart a Linux computer with a touchscreen.

This meant that it had to be about as difficult to install and configure as a lamp, and a network of these devices had to be about as difficult to install as a group of lights on the same circuit.

The way that the designers solved this problem was by including the whole service in every device. Every device worked out of the box standalone: if you take it out of the box, hang it on a wall and plug it in, you can start scheduling that conference room. You can finish right there, but if you want to allow people to schedule conference rooms from their desk, then each device has a web server and a database on it.

Photo: Haiyan Zhang



You can point your web browser to it and start scheduling rooms with it.

Where it gets interesting is when there are multiple Roomwizard devices. Each one is a standalone device with a complete version of the service, but the devices can be “introduced” to each other and they will automatically synchronize with each other, negotiating all appropriate schedule synchronization from then on, effectively becoming clones. Every one will have the complete schedule for every other one, and any can be accessed to get the same service.

It doesn't make it any less of a service, since commands given to one avatar may initiate a scheduling synchronization in every other device. It does, however, make it a service that has no center.



Next, I want to talk about the Nabaztag, that famous Wifi Rabbit. What's interesting about the Nabaztag is that it was consciously designed as a kind of hollow avatar, an empty shell with a range of expressive functions that could serve as the input and output for arbitrary services. The developers did this on purpose. They wanted to design a range of service avatars, but they did not have a strong idea of what avatar would be best to develop, so they developed a product that could be an avatar for a wide range of services. Maybe it wouldn't be a great avatar for most things, but the developers felt that by putting it out in the world as a kind of a blank slate, maybe they'd find one that worked well.

In a sense mobile phones have a kind of hollow avatar quality, too—after all, there are now 85,000 different ways that people have thought of using an iPhone—the difference is that phones are first and foremost avatars for several specific services: phone call, text messaging, lightweight photography and casual internet browsing. These services shape phones in specific ways: they have to be portable, sturdy, they have to have a screen, etc. The Nabaztag's designers decided to see what happened if you had a different set of tangible interaction constraints first, and then see what services fit.



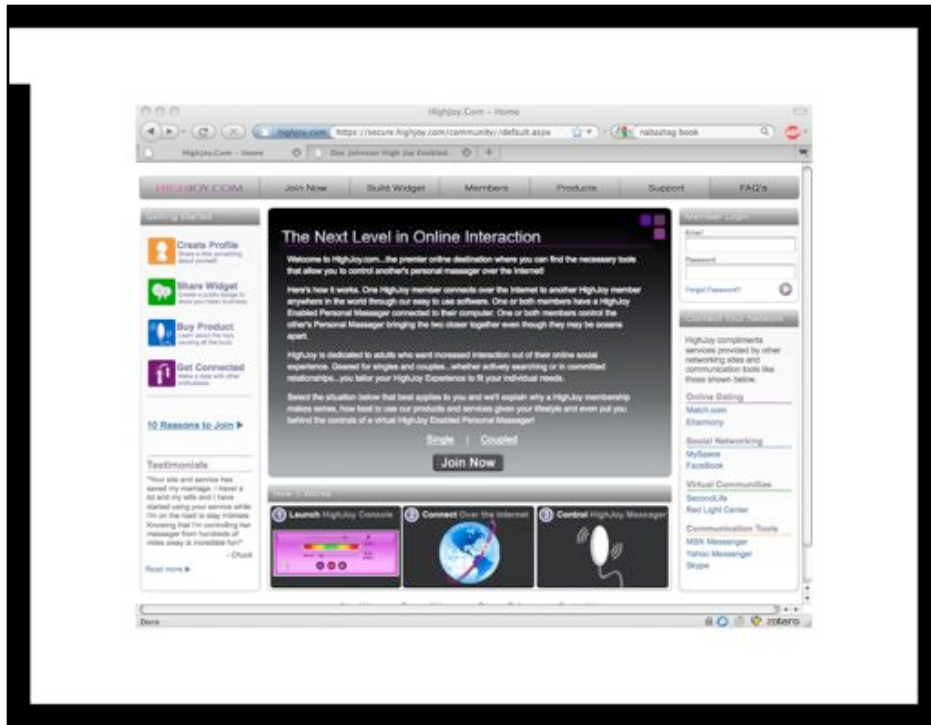
I think that the most successful of these experiments was the children's audio book reading service. Violet, Nabaztag's manufacturer made a deal with a children's book publisher to produce RFID enabled children's books. When a child waved the book in front of an RFID-enabled Nabaztag's RFID reader, it would trigger the downloading of an MP3 audio stream and the Rabbit would read the book through its internal speaker.



Violet, Nabaztag's manufacturer, went bankrupt earlier this year, but before they went under they put out a product that I believe was directly inspired by that specific service. They put out a simple USB RFID reader called the Mir:ror. I think that the name was not accidental: they had clearly begun to think about digital services as reflecting RFID-enabled objects, and vice versa.

In a sense, the Nabaztag experiment was a success and they found a more specific service/product relationship they wanted to pursue, but it came too late to save the company.

The founder of the company, by the way, recently started an Internet of Things company, so it'll be interesting to see where he goes with these ideas.



Finally, I'd like to talk about one of the most tangible interfaces I know. This is highjoy.com. This is a dating site and, interestingly, it's also a site for couples who are physically far apart. That makes it unusual already, since uniting physical separation and dating are typically not dealt in the same website. One is usually a kind of social network and the other is some kind of telepresence service. What highjoy realized is that they have one thing in common, which is sex.

So highjoy created a service that let people have a kind of sex remotely. The service is sexual gratification, transmitted through hardware avatars.



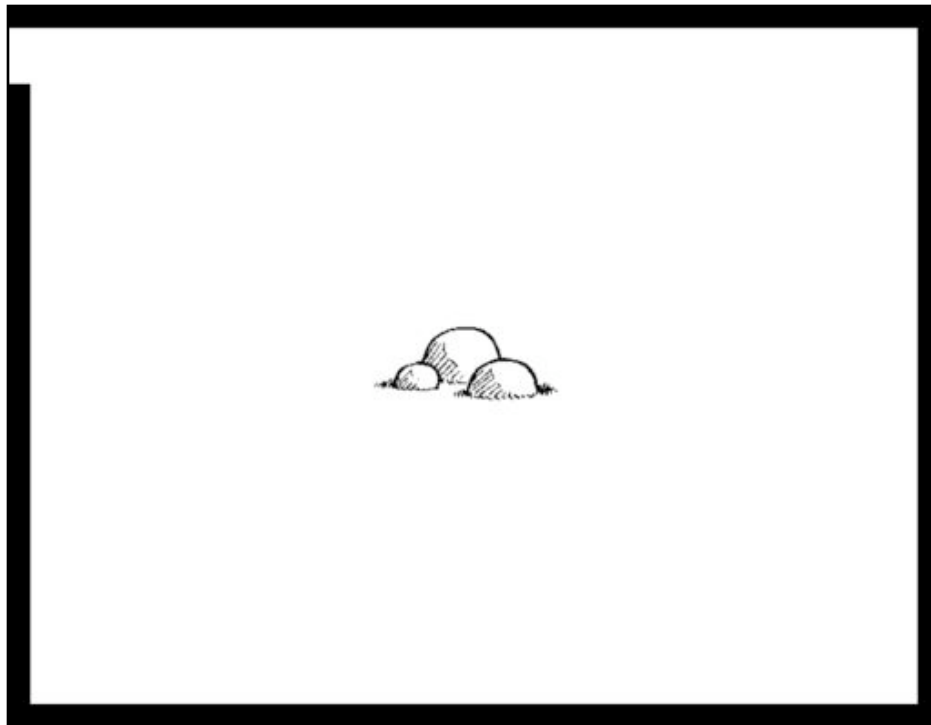
There are two interface products, which look pretty much how you'd expect them to look. One is for women and one is for men.

They work in two ways: first, the user of one of these can use the buttons on the remote control to manipulate their product. In this way, it's a straightforward standalone product, like many others on the market.



Highjoy's innovation is that they allow the devices to be connected to the highjoy service and controlled remotely. That way two people who have met through the dating site can go on a virtual date with highjoy's system: they can talk to each other over text messaging or chat over Skype and, when things start getting intimate, they can grab each other's consoles and create a low-tech simulation of sexual intimacy. Couples in committed relationship but who are physically separated can do the same thing, which is why the site provides that service as well.

The underlying service is the transmission of low-resolution sexual information, and the interface devices are the physical representations of the service. Unlike the Nabaztag or a phone, for example, there's no question about what kind of information the service will carry, so the devices can be very precise and direct.



There many significant interaction design issues in creating physical avatars of services. Today I want to leave you with three.

First, how much of the functionality of the device is local versus remote. When the cloud goes down, as it inevitably occasionally will (ask T-Mobile), how much functionality does the device still have? When the data network goes down, does the product become useless? The Roomwizard's designers made their product still do most of its job if there was no network, the highjoy products certainly work offline, and the Nabaztag's designers wanted their device to at least be decorative and have an entertaining offline mode, but the designers of ATMs decided that ATMs should be useless without the network. This split now becomes a specific design decision.

Second, what's the business model going to be? Is the avatar product going to be bought, rented or subscribed to? In the old days, Bell phones were not owned, but rented, and this decision greatly affected their design. New car designs are being developed to cater explicitly to models like City Carshare. If a beloved Clickable becomes a beloved personal item, and a parent forgets to pay the rent on it, will it be taken away? Could I subscribe to a Nabaztag-like device, and receive new hardware with new capabilities on a regular basis?

Finally, in manifesting abstract services as physical avatars, the physical specifics become much more important than when you're designing for a general-purpose device like a web browser or a phone. How big is going to be? Where is going to go? The strength of digital services is that they can manifest as multiple avatars, but each of those avatars is a specific product. The product is then not just a way to provide access to something else in the cloud, in a very literal sense it IS the cloud. And how is that shaped?



Mike Kuniavsky
mikek@thingm.com

Thank you.